CBP-Based Fast Mode Decision for Scalable Video Coding

Bo-Yin Chou, Yu-Hung Chen, Chung-Ping Lai and Chia-Hung Yeh[†]
Department of Electrical Engineering, National Sun Yat-Sen University, Kaohsiung, 804 Taiwan R.O.C.
[†]E-mail: yeh@mail.ee.nsysu.edu.tw

***Abstract*—In order to speedup the encoding processing of Scalable Video Coding (SVC), a CBP (Coded Block Pattern)-based fast mode decision algorithm is proposed in the paper. The CBP value is the syntax used at each Macroblock (MB) header to indicate whether an MB contains residual information or not. By the CBP values and MB modes of adjacent MB in Enhancement Layer (EL) and the co-located Base Layer (BL) MB mode, the proposed algorithm can exclude the invalid modes for the mode prediction of the current MB in EL. Experimental results show that the proposed algorithm achieves more than 76% EL Motion Estimation (ME) time-saving with ignorable PSNR degradation and bit increase when compared to JSVM 9.12.**

I. Introduction

Scalable video coding proposed in [1-3] is an extended version of H.264/AVC. Compared to other existing coding standards, SVC can use only a single bitstream to provide multiple temporal, spatial and quality scalabilities with high compression efficiency, but it also results in high computational complexity. The basic structure of SVC is one H.264/AVC-compatible BL and multiple ELs. Temporal scalability is archived by the hierarchical B frame coding structure that allows extracting a subset video with different frame rates from a single bitstream. For the spatial scalability, the BL contains a down-sampled version of each coded frame. The ELs are coded based on BL frame and previously encoded EL frame. Quality scalability provides multiple quality levels. Similar with spatial scalability, quality scalability also contains a BL and several ELs, which are coded at the same spatial and temporal resolution. In order to improve the coding efficiency, the above scalabilities are achieved by the pyramid coding structure in combination with adaptive inter-layer prediction, resulting in large computing complexity. In SVC, motion estimation with the variable block-size is adopted to reduce the temporal residue between frames. In [4-8], several fast mode decision algorithms are proposed to reduce the complexity of encoder.

In the previous algorithms, MB modes of EL are generally predicted from BL information. In [4], the fast mode decision algorithm of quality scalability is proposed by the fact that the MB in EL will have finer partition than their co-located MBs in BL. Similar principles are further used to reuse the rate-distortion cost of the BL for spatial scalability [5]. In [6], the information of temporal level in BL is employed as the indicator for the mode selection in different temporal layers. In [7], the authors analyze the distribution of the best modes according to various CBP values and use the information to determine the MB mode in H.264/AVC. In [8], the authors analyzed the accuracy of the initial search point for motion estimation in EL and explored the mode relation between BL and EL according to different QPs.

In this paper, a CBP-based fast mode decision algorithm is proposed to speedup the encoding process of SVC. By referring to the CBP values of adjacent MBs in EL the proposed algorithm excludes the invalid modes to be tested. Furthermore, the mode of the EL frames in the largest temporal level of GOP can be determined by the temporal relativity mode selection method. Experimental results show that the proposed algorithm can save up to 76% EL ME time-saving with JSVM 9.12 [9] and outperforms other existing algorithms.

The rest of this paper is organized as follows: Section 2 explains analyses of CBP and the largest temporal level information. Based on the aforementioned analyses, Sec. 3 presents the proposed CBP-based fast mode decision algorithm. Experimental results are shown in Sec. 4 Concluding remarks are given in Sec. 5.

II. Analyses of CBP Characteristics and Largest Level Temporal Information

A. *The Analysis of CBP Characteristics*

The CBP value is a syntax used at each H.264/AVC MB header. The CBP value indicates that the six 8×8 blocks, including 4 luma and 2 chroma blocks, may contain non-zero transform coefficients [10]. The bit structure of CBP is shown in Fig. 1. *CBPLuma* and *CBPChroma* are defined as

$$CBPLuma = \text{CBP value} \ \% \ 16, \tag{1}$$
$$CBPChroma = \text{CBP value} / 16. \tag{2}$$

For each of four 8×8 luma block of MB, the *CBPLuma* could be one of the following cases:

1. All the 4×4 luma blocks in the 8×8 luma block have zero coefficients.
2. One or more the 4×4 luma blocks in each 8×8 luma block require residual coding.

For example, *CBPLuma* = 0 (0000) means all the luma blocks are unnecessary coded and *CBPLuma* = 5 (0101) means the

09-0100790083©2009 APSIPA. All rights reserved.

block Y_0 and Y_2 have nonzero coefficients. The meaning of *CBPChroma* is explained as:

1. If *CBPChroma* =0 (00), all the chroma blocks don't need to be coded.
2. If *CBPChroma* =1 (01), the DC coefficients of chroma blocks is non-zero value and all the AC coefficients of chroma blocks are equal to zero.
3. If *CBPChroma* =2 (10), one or more AC coefficients and zero or more DC coefficients of chroma blocks are non-zero value.

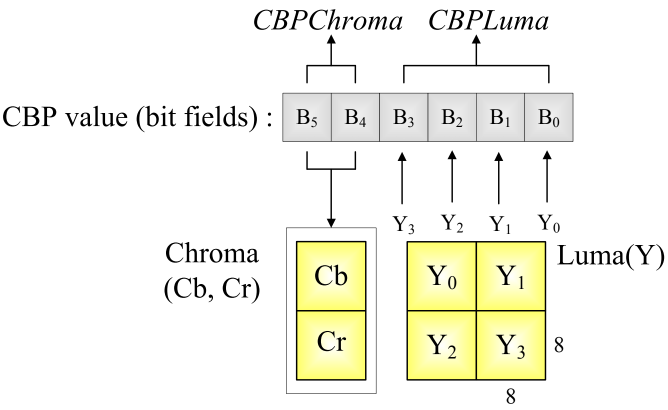

Fig. 1 The bit structure of CBP

In summary, the distribution of the CBP locates between 0(000000) to 47(101111), which uses to describe the need of residual coding for luma or chroma blocks.

In [7], the authors analyze the distribution of the best modes with respect to various CBP values. It shows that skip, Inter16×16, Inter16×8 and Inter8×16 are the best mode if zero or one luma block requires residual coding and InterP8×8 is the best mode when all luma blocks are non-zero. The InterP8×8 includes Inter8×8, Inter8×4, Inter4×8 and Inter4×4. Therefore, CBP is a good indicator, which can be used to determine coding mode fast.

B. *Analysis of the Largest Temporal Level Information*

As illustrated in Fig. 2, the frame in the largest temporal level of a GOP is highly correlated with its previous/next frames. Table I shows the statistical results of the mode of the current MB (MB_mode_t) with respects to the previous co-located MB mode (MB_mode_{t-1}) and the next MB mode (MB_mode_{t+1}). $P(Match|MB\_mode_{t-1})$ is conditional probability given $MB\_mode_{t-1}=MB\_mode_{t+1}$. *Match* means the current MB mode equal to the previous/next MB mode or skip mode in different QPs.

In Table I, the probability of the current MB mode equal to previous/next MB's would be 96.1% if previous/next MB modes are the same. Therefore, we can estimate the current MB mode by analyzing the previous/next co-located MB modes for the frames in the largest temporal level. As mentioned above, total coding time can be reduced significantly.

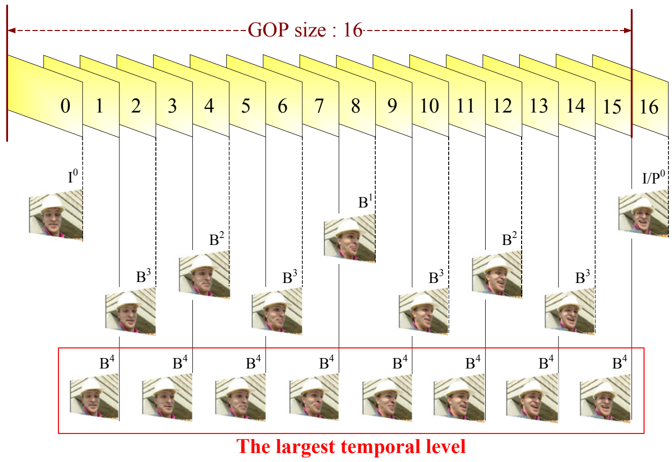

Fig. 2 The hierarchical B frame coding structure

TABLE I
THE STATISTICS $P(Match\,|\,MB\_mode_{t-1} = MB\_mode_{t+1})(\%)$

Sequence	QP 10	QP 20	QP 30	QP 40	QP 50	Avg.
Foreman	96.9%	96.9%	95.5%	93.1%	91.8%	94.8%
Container	99.9%	99.9%	99.9%	99.7%	98.1%	99.5%
Akiyo	99.7%	99.7%	99.7%	99.5%	97.9%	99.3%
Stefan	92.6%	92.7%	91.8%	88.8%	87.9%	90.8%
Table	96.6%	97.0%	96.5%	95.3%	94.8%	96.0%
Avg.	**97.1%**	**97.2%**	**96.7%**	**95.3%**	**94.1%**	**96.1%**

III. PROPOSED CBP-BASED FAST MODE DECISION ALGORITHM

As the observation in Sec. 2, this paper develops a CBP-based fast mode decision algorithm, which combines the temporal relativity mode selection method. When the frames are not in the largest temporal level of GOP in EL, the proposed algorithm determine the required number of test coding modes in EL shown in Fig. 4. CBP value is adopted to determine the MB complexity and the candidate modes required can be reduced. For the frames in the largest temporal level of GOP in EL, the temporal relativity mode selection method is used to decide the test mode.

A. *CBP-Based Fast Mode Decision*

The information of adjacent MBs, such as CBP value and partition mode, is a good indicator to predict the candidate modes of current MB in the homogenous field. Moreover, motion information of each layer can be reused to predict the MB mode for the following ELs [11]. Therefore, the information of adjacent and BL can be used to predict the current MB mode in EL.

In Fig. 4, when zero or only one luma block of both above and left MB is necessary to be encoded, the direct mode, *BL_mode* and *ad_mode*, are selected as the candidate coding mode. *BL_mode* means the co-located BL MB mode and the *ad_mode* means the extended direction of adjacent MB mode in EL. For examples, as shown in Fig. 3, when the above and left MB modes are both Inter16×8, the *ad_mode* is predicted as Inter16×8. When the above MB mode is Inter8×8 and the left MB mode is Inter16×8, the *ad_mode* is predicted as

Inter8×8 and so on. If the luma blocks of adjacent MBs are non-zero value, it means the current MB is complicated and Inter16×16, Inter16×8 and Inter8×16 are insufficient. Therefore, InterP8×8 should be executed to improve the accuracy of the estimation.

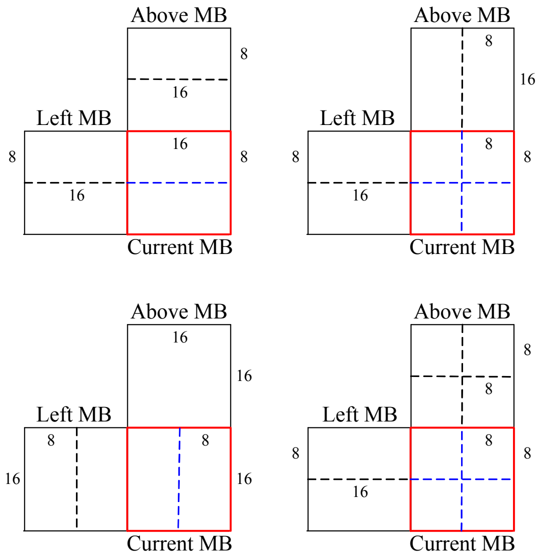

Fig. 3 The adjacent MB mode

When the CBP value doesn't satisfy the above two cases, it is difficult to determine the complexity of the current MB. All modes should be checked to obtain acceptable results.

B. *The temporal relativity mode selection method*

In order to reduce ME coding time in EL, the temporal relativity mode selection method decides the candidate mode according to the largest temporal level information. From Table I, we can know that if $MB\_mode_{t-1} = MB\_mode_{t+1}$, the current MB mode is almost equal to the previous/next MB mode for the frames in the largest temporal level of GOP. Therefore, when $MB\_mode_{t-1} = MB\_mode_{t+1}$, the MB_mode_{t-1} and direct mode can be immediately selected as the candidate mode.

IV. Experimental Results

The proposed algorithm and Lin's algorithm [8] are both simulated on the reference software JSVM 9.12. The simulation setting is shown as Table II. Four-layer simulation tests five benchmarks, including *Foreman*, *Table, Container*, *Akiyo* and *Soccer* with CIF format. QP value setting for BL/EL, GOP size and frame number are shown in Table II. The bitrate, PSNR, total time, ME time, EL ME time and saving rate of selecting block mode in EL are applied for evaluation. We implemented Lin's algorithm without MODE_SR in JSVM 9.12 since MODE_SR has been removed. The motion search method in Lin's algorithm is also removed since we want to focus on the comparison of mode decision algorithm. *TS*, *EL_ME_TS*, Δ*PSNR*, Δ*bitrate* and *SR* represent the gains in total time-saving, EL ME time-saving, PSNR, bitrate and saving rate of selecting block mode in EL, respectively, and defined as

$$TS = \frac{Total\ time_{proposed} - Total\ time\ of\ JSVM\ 9.12}{Total\ time\ of\ JSVM\ 9.12} \times 100\%, \quad (3)$$

$$EL\_ME\_TS = \frac{ME\ time_{proposed}\ in\ EL - ME\ time\ of\ JSVM\ 9.12\ in\ EL}{ME\ time\ of\ JSVM\ 9.12\ in\ EL} \times 100\%, \quad (4)$$

$$\Delta PSNR = PSNR_{proposed} - PSNR\ of\ JSVM\ 9.12, \quad (5)$$

$$\Delta bitrate = \frac{bitrate_{proposed} - bitrate\ of\ JSVM\ 9.12}{bitrate\ of\ JSVM\ 9.12} \times 100\%, \quad (6)$$

$$SR = \frac{selecting\ block\ mode\ number_{proposed} - selecting\ block\ mode\ number_{JSVM\ 9.12}}{selecting\ block\ mode\ number_{JSVM\ 9.12}} \times 100\%. \quad (7)$$

From Tables III and IV, the propose algorithm provides 75% EL ME time-saving in average compared to JSVM 9.12 and outperform Lin's algorithm with ignorable PSNR degradation and bit increase. Tables V~VII show the performance comparisons in terms of the total time, EL ME time, PSNR, bitrate and saving rate of selecting block mode in EL in small QP-delta ($QP_{EL} - QP_{BL}$). The QP of BL is set to 40. The QPs of ELs are set to 35, 30 and 25, respectively. The resolution of testing sequences is CIF format. In small QP-delta, the proposed algorithm still can reduce more than 76% EL ME time-saving with negligible PSNR degradation and bit increase compared with JSVM 9.12. From Tables III~VI, we can confirm that the efficiency of the proposed algorithm is better than Lin's algorithm whether in large QP-delta or in small QP-delta. From Table VII, we can find that the proposed algorithm can reduce more than 67% selecting block mode compared with JSVM 9.12. That is the reason the proposed algorithm can reduce the MB time in EL significantly. In summary, the proposed algorithm always has better coding performance compared with other existing algorithms.

V. Conclusions

In this paper, a CBP-based fast mode algorithm is proposed to reduce the computational complexity of SVC encoder. The proposed algorithm determines the complexity of the current MB using CBP values of adjacent MBs. Then, according to different CBP values, the MB prediction modes of EL can be estimated by the MB partition of co-located MB in BL or extended direction of the adjacent MB. Moreover, the largest temporal level information is analyzed in this paper. Based on the aforementioned analysis, the temporal relativity mode selection method is used to select the best inter mode of the largest temporal level frame in EL. Since CBP value, adjacent and BL information is already computed before mode estimation process, few extra computational time is required for the CBP-based fast mode decision algorithm. Experimental results show that the proposed algorithm can reduce encoding time significantly.

Acknowledgment

The authors would like to thank the National Science Council of the Republic of China for financially supporting this research under Contracts No. NSC-98-2221-E-110-057.

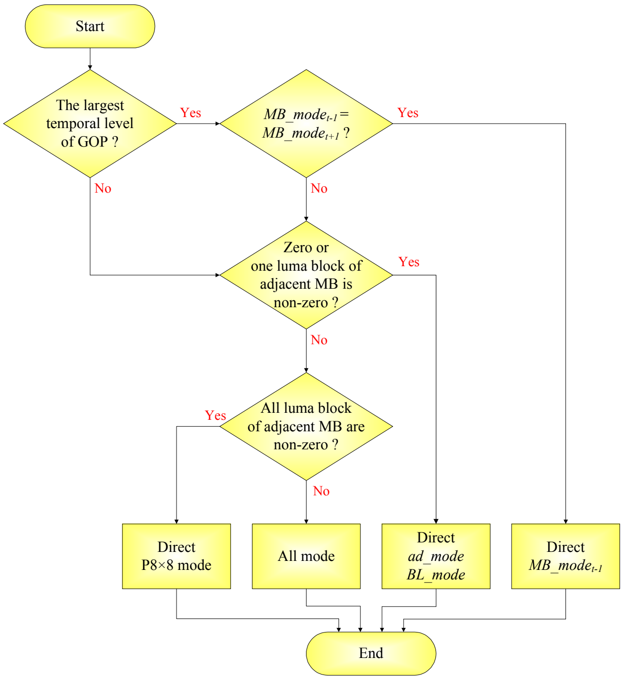

Fig. 4 Flowchart of CBP-based fast mode decision

TABLE II
SIMULATION SETTING OF FOUR LAYERS

Resolution	CIF
Frame Rate	30 Hz
The Number of Frames	100
QP_B	40
QP_{E1}, QP_{E2}, QP_{E3},	30, 20, 10
Encoder Configuration	MV search range : ±16 pixels GOP size : 16 RDO : enabled Number of reference frame : 1

TABLE III
THE COMPARISON OF *TS* AND *EL_ME_TS*

Sequence	*TS* (%)		*EL_ME_TS* (%)	
	Lin [8]	Proposed	Lin [8]	Proposed
Foreman	-47.43%	-64.30%	-54.27%	-73.58%
Table	-46.94%	-63.49%	-53.66%	-72.55%
Container	-50.74%	-69.15%	-57.35%	-78.14%
Akiyo	-50.59%	-71.15%	-57.21%	-80.49%
Soccer	-48.18%	-65.43%	-54.80%	-74.27%
Avg.	**-48.78%**	**-66.70%**	**-55.46%**	**-75.81%**

TABLE IV
THE COMPARISON OF △*PSNR* AND △*BITRATE*

Sequence	△*PSNR* (dB)		△*bitrate* (%)	
	Lin [8]	Proposed	Lin [8]	Proposed
Foreman	0	0	0.65%	0.38%
Table	-0.01	-0.01	0.32%	0.41%
Container	-0.01	0	1.30%	0.08%
Akiyo	0	0	0.64%	0.05%
Soccer	0	0	1.16%	0.34%
Avg.	**-0.004**	**-0.002**	**0.81%**	**0.25%**

TABLE V
THE COMPARISON OF *TS* AND *EL_ME_TS* IN SMALL QP-DELTA

Sequence	*TS* (%)		*EL_ME_TS* (%)	
	Lin [8]	Proposed	Lin [8]	Proposed
Foreman	-42.58%	-65.02%	-48.81%	-74.45%
Table	-42.24%	-64.45%	-48.39%	-73.58%
Container	-43.62%	-70.57%	-49.35%	-79.69%
Akiyo	-43.51%	-71.40%	-49.34%	-80.80%
Soccer	-42.98%	-63.54%	-49.16%	-73.99%
Avg.	**-42.99%**	**-67.00%**	**-49.01%**	**-76.50%**

TABLE VI
THE COMPARISON OF △*PSNR* AND △*BITRATE* IN SMALL QP-DELTA

Sequence	△*PSNR* (dB)		△*bitrate* (%)	
	Lin [8]	Proposed	Lin [8]	Proposed
Foreman	-0.02	-0.03	1.21%	1.82%
Table	-0.03	-0.02	2.98%	1.66%
Container	-0.00	-0.00	0.01%	0.28%
Akiyo	-0.01	-0.01	0.14%	0.45%
Soccer	-0.02	-0.04	0.35%	1.46%
Avg.	**-0.016**	**-0.020**	**0.94%**	**1.13%**

TABLE VII
THE COMPARISON OF *SR* IN SMALL QP-DELTA

Sequence	Lin [8]	Proposed
Foreman	-48.42%	-64.20%
Table	-48.84%	-63.12%
Container	-46.48%	-71.30%
Akiyo	-46.42%	-74.54%
Soccer	-49.10%	-62.66%
Avg.	**-47.85%**	**-67.16%**

REFERENCES

[1] T. Wiegand, G. Sullivan, J. Reichel, H. Schwarz, and M. Wien, "Joint draft 10 of SVC amendment," *ISO/IEC JTC1/SC29/WG11 and ITU-T SG16 Q.6, JVT-W201*, April, 2007.

[2] H. Schwarz, D. Marpe, and T. Wiegand, "Overview of the scalable video coding extension of the H.264/AVC standard," *IEEE Transactions on Circuits and Systems for Video Technology (CSVT)*, vol. 17, no. 9, pp.1103-1120, 2007.

[3] C. A. Segall and G. J. Sullivan, "Spatial scalability within the H.264/AVC scalable video coding extension," *IEEE Transactions on Circuits and Systems for Video Technology (CSVT)*, vol. 17, no. 9, pp.1121-1135, 2007.

[4] H. Li, Z.-G. Li, and C. Wen, "Fast mode decision for coarse grain SNR scalable video coding," *in Proceedings of IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 545-548, 2006.

[5] H. Li, Z.-G. Li, and C. Wen, "Fast mode decision for spatial scalable video coding," *in Proceedings of IEEE International*

Symposium on Circuits and Systems (ISCAS), pp. 3005-3008, 2006.

[6] H. Li, Z.-G. Li, and C. Wen, "Fast mode decision algorithm for inter-frame coding in fully scalable video coding," *IEEE*

[7] *Transactions on Circuits and Systems for Video Technology (CSVT)*, vol. 16, no. 7, pp.889-895, 2006.

[8] B.-Y. Chen and S.-H Yang, "Using H.264 coded block patterns for fast inter-mode selection," *in Proceedings of IEEE International Conference on Multimedia and Expo (ICME)*, pp. 721-724, 2008.

[9] H.-C. Lin, W.-H. Peng, H.-M. Hang, and W.-J. Ho, "Layer-adaptive mode decision and motion search for scalable

[10] video coding with combined coarse granular scalability (CGS) and temporal scalability," *in Proceedings of IEEE International Conference on Image Processing (ICIP)*, pp. 289-292, 2007.

[11] J. Vieron, M. Wien, and H. Schwarz, "JSVM 10 software," *ISO/IEC JTC1/SC29/WG11 and ITU-T SG16 Q.6, JVT-W201*, April, 2007.

[12] ISO/IEC International Standard 14496-10, Information technology – Coding of audio-visual objects – Part 10: Advanced Video Coding, second edition, 2004.

[13] Q. Dai, D. Zhu, and R. Ding, "Fast mode decision for inter prediction in H.264," *in Proceedings of IEEE International Conference on Image Processing (ICIP)*, pp.119-122, 2004.